ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Journal of Computer and System Sciences 69 (2004) 435–447

JOURNAL OF COMPUTER AND SYSTEM SCIENCES

http://www.elsevier.com/locate/jcss

Cell-probe lower bounds for the partial match problem

T.S. Jayram,[a] Subhash Khot,[b,1] Ravi Kumar,[a,*] and Yuval Rabani[c,2]

[a] *CS Principles and Methodologies, IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA*

[b] *Department of Computer Science, Princeton University, Princeton, NJ 08544, USA*

[c] *Computer Science Department, Technion, Israel Institute of Technology, Technion City, Haifa 32000, Israel*

Received 14 September 2003; accepted in revised form 13 April 2004

Available online 2 June 2004

Abstract

Given a database of n points in $\{0,1\}^d$, the *partial match* problem is: In response to a query x in $\{0,1,*\}^d$, is there a database point y such that for every i whenever $x_i \neq *$, we have $x_i = y_i$. In this paper we show randomized lower bounds in the cell–probe model for this well-studied problem (Analysis of associative retrieval algorithms, Ph.D. Thesis, Stanford University, 1974; The Art of Computer Programming; Sorting and Searching, Addison-Wesley, Reading, MA, 1973; SIAM J. Comput. 5(1) (1976) 19; J. Comput. System Sci. 57(1) (1998) 37; Proceedings of the 31st Annual ACM Symposium on Theory of Computing, 1999; Proceedings of the 29th International Colloquium on Algorithms, Logic, and Programming, 1999).

Our lower bounds follow from a near-optimal asymmetric communication complexity lower bound for this problem. Specifically, we show that either Alice has to send $\Omega(d/\log n)$ bits or Bob has to send $\Omega(n^{1-o(1)})$ bits. When applied to the cell–probe model, it means that if the number of cells is restricted to be $\text{poly}(n,d)$ where each cell is of size $\text{poly}(\log n, d)$, then $\Omega(d/\log^2 n)$ probes are needed. This is an exponential improvement over the previously known lower bounds for this problem obtained by Miltersen et al. (1998) and Borodin et al. (1999).

*Corresponding author.

E-mail addresses: jayram@almaden.ibm.com (T.S. Jayram), khot@cs.princeton.edu (S. Khot), ravi@almaden.ibm.com (R. Kumar), rabani@cs.technion.ac.il (Y. Rabani).

[1] Supported by Prof. Sanjeev Arora's David and Lucile Packard Fellowship, NSF Grant CCR-0098180, and an NSF ITR Grant. Part of this work was done while the author was visiting the IBM Almaden Research Center and IBM T.J. Watson Research Center.

[2] Work at the Technion supported by BSF Grant number 99-00217, by ISF Grant number 386/99, by IST contract number 32007 (APPOL), and by the Fund for the Promotion of Research at the Technion. Part of this work was done while the author was visiting the IBM Almaden Research Center.

doi:10.1016/j.jcss.2004.04.006

Our lower bound also leads to new and improved lower bounds for related problems including a lower bound for the ℓ_∞ c-nearest neighbor problem for $c<3$ and an improved communication complexity lower bound for the exact nearest neighbor problem.

Keywords: Cell-probe model; Partial match problem; Nearest neighbor problem; Asymmetric communication complexity

1. Introduction

Given a database of points in a space and a query, the decision version of the *nearest neighbor* problem is to determine if there is a database point that is "close" to the query. Here, the notion of closeness depends on the underlying space and also on whether a close, but not necessarily the closest data point suffices. The goal is to preprocess the database and create a static data structure that will support efficient search. The complexity measures are the size of the data structure (usually, a table) and the number of probes into the table.

The nearest neighbor problem is a fundamental problem in computational geometry with numerous applications and many efficient algorithms are known for low-dimensional Euclidean spaces. However, these algorithms become inefficient in terms of query time and/or space requirements as the dimension grows—this is the so-called *curse of dimensionality*. This prompted the quest for very efficient (and possibly randomized) algorithms for the approximate nearest neighbor problem in Euclidean/Hamming spaces and the last few years have seen tremendous progress made in this direction [Kle97,IM98,KOR00]; also supporting this line of study was the evidence for the curse of dimensionality for these spaces provided by the recent lower bounds of [BOR99,CCGL99,BR00]. However, for many nearest neighbor problems, there is still a huge gap between the upper and lower bounds. In fact, for many of these problems, the gap is exponential.

1.1. Our results

In this paper we consider an instantiation of the nearest neighbor problem—the *partial match* problem—where, in response to a query x in $\{0,1,*\}^d$, decide if there is a database point y such that for every i whenever $x_i \neq *$, we have $x_i = y_i$. This problem has been investigated for the last few decades (see, for example, [Riv74,Knu73]), but all solutions discovered so far have either essentially $\Omega(n)$ query time or $2^{\Omega(d)}$ space. This leads to the widely believed conjecture that this problem also suffers from the curse of dimensionality (see [BOR99,BR00,CIP02]). We give supporting evidence for this conjecture by proving essentially optimal communication complexity lower bounds for this problem, and thus substantially improving the existing lower bounds in the cell–probe model.

Specifically, we show that in the cell–probe model, if the number of cells is restricted to be $\mathrm{poly}(n,d)$ where each cell is of size $\mathrm{poly}(\log n, d)$, then $\Omega(d/\log^2 n)$ probes are needed; this is an exponential improvement over the bounds in [MNSW98,BOR99]. Our lower bound follows from an asymmetric two-sided error randomized communication complexity lower bound for this

problem. Specifically, we show that either Alice has to send $\Omega(d/\log n)$ bits or Bob has to send $\Omega(n^{1-o(1)})$ bits.

There are several consequences of this basic lower bound and the most notable ones include: (1) a near-optimal communication complexity lower bound and a cell–probe lower bound for the ℓ_∞ c-nearest neighbor problem for $c<3$; this addresses the presumed hardness used by Indyk [Ind01] and (2) a near-optimal communication complexity lower bound for the Hamming nearest neighbor problem; this strengthens the bound on Bob's communication obtained by Barkol and Rabani [BR00].

1.2. Related work

As mentioned earlier, the problem of partial match has been investigated for quite a while. The first non-trivial result for this problem was obtained by Rivest [Riv74,Riv76], who showed that for $d \leqslant 2\log n$, the "exhaustive storage" solution can somewhat be improved. Recently, Charikar et al. [CIP02] presented two algorithms with the following query-time-space tradeoffs: $n\exp(O(d\log^2 d\sqrt{c/\log n}))$ space and $O(n/2^c)$ query time for any c and nd^c space and $O(dn/c)$ query time for any $c \leqslant n$. If the number of cells is restricted to be at most $\mathrm{poly}(n,d)$ where each cell is of size $\mathrm{poly}(\log n, d)$, the previously known cell–probe lower bounds were $\Omega(\sqrt{\log d})$ due to Miltersen et al. [MNSW98] and $\Omega(\log d)$ due to Borodin et al. [BOR99].

The nearest neighbor problem in d-dimensional Euclidean space or the d-dimensional Hamming cube is also a well-studied problem. Note that the exact version of nearest neighbor in the Hamming cube can be solved trivially by a single probe into a table of 2^d cells, each containing d bits. The best algorithms for the exact nearest neighbor problem in Euclidean space take $\mathrm{poly}(d, \log n)$ query time and need $n^{\Theta(d)}$ space (see, for instance, [Mei93]). For the approximate version of the problem, the best known algorithms are randomized and make $O(\log\log d)$ probes to a table of size $\mathrm{poly}(n,d)$ [IM98,KOR00]. The current best known cell–probe randomized lower bound for the exact nearest neighbor problem (in both the Hamming and the Euclidean cases) is $\Omega(d/\log n)$ for a table of size $\mathrm{poly}(n,d)$ [BOR99,BR00]; in a less general yet reasonable model, Beame and Vee [BV02] showed a lower bound of $\Omega(d\sqrt{\log d/\log\log d})$. For the approximate nearest neighbor problem in the Hamming cube, Chakrabarti et al. [CCGL99] showed a lower bound of $\Omega(\log\log d/\log\log\log d)$ and Liu [Liu03] has recently improved this to $\Omega(d^{1-o(1)})$; these lower bounds, however, apply only to deterministic algorithms. Very recently, Chakrabarti and Regev [CR03] have obtained a lower bound of $\Omega(\log\log d/\log\log\log d)$ even for randomized algorithms.

For the nearest neighbor problem under the ℓ_∞ norm, Indyk [Ind01] gave a 3-approximation algorithm that makes $O(d\log n)$ probes into a table of size $O(n^{1+\log d})$ and an $O(\log\log d)$-approximation algorithm that uses an essentially linear-sized table. This paper also shows that for factors below 3, this problem is as hard as the partial match problem. No lower bound better than $\Omega(\log d)$ was known before for the ℓ_∞ nearest neighbor problem.

1.3. Organization

In Section 2, we provide a brief description of the lower bound technique that we use. In Section 3, we will obtain a cell–probe lower bound for an intermediate problem called Intersect All.

Finally, in Section 4, we use this lower bound to demonstrate near-optimal lower bounds for partial match, exact nearest neighbor, ℓ_∞ nearest neighbor, and other related problems.

2. Background

2.1. The cell–probe model

The *cell–probe model*, formulated by Yao [Yao81], is a model for studying the complexity of data structure problems. In a (static) data structure problem, we have a domain $\mathcal{Y}$ of possible *databases*, a domain $\mathcal{X}$ of possible *queries*, and a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, where $f(X, Y)$ is the answer to query $X \in \mathcal{X}$ about database $Y \in \mathcal{Y}$. A solution in the cell–probe model with parameters s, b, and t is a method for storing any database as a data structure in the memory of a random access machine such that: s denotes the *number of cells* in the data structure, b denotes the maximum *cell size* in terms of the number of bits in the cells, and t denotes the maximum *number of probes* made into the data structure for any query. An important technique to obtain lower bounds in the cell–probe model is via asymmetric communication complexity [Ajt88,Mil94]. For more details on the cell–probe model, see the survey by Miltersen [Mil99].

2.2. Asymmetric communication complexity

For a data structure problem with $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$, consider the asymmetric communication complexity model with two players, Alice and Bob. Alice gets $X \in \mathcal{X}$ and Bob gets $Y \in \mathcal{Y}$ and their goal is to compute $f(X, Y)$. The complexity measure is the total number of bits communicated by each player—an $[a, b]$-*protocol* is one in which Alice sends a bits and Bob sends b bits. A randomized protocol is said to have two-sided error ε if it computes the function on every input correctly with probability at least $1 - \varepsilon$.

Lower bounds on (randomized) asymmetric communication complexity lead to (randomized) lower bounds in the cell–probe model.

Lemma 1 (Miltersen [Mil94]). *For any function, if there is a* (*randomized*) *solution in the cell–probe model with parameters* s, b, *and* t, *then there is a* (*randomized*) $[t\lceil \log s \rceil, tb]$-*protocol for the corresponding communication problem.*

It therefore suffices to show lower bounds on the (randomized) asymmetric communication complexity. To accomplish this, we use the *richness* technique due to Miltersen et al. [MNSW98], which we describe below. This presentation is slightly different from the version present in [MNSW98], but is more convenient for us to work with.

2.3. The richness technique

It will be convenient to view any function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ via its *function matrix* M_f. This is a 0–1 matrix whose rows are indexed by inputs $X \in \mathcal{X}$ to Alice and the columns are indexed by inputs $Y \in \mathcal{Y}$ to Bob and is such that $M_f(X, Y) \stackrel{\text{def}}{=} f(X, Y)$. This is also helpful when dealing with

deterministic protocols where $f(X, Y)$ denotes the output of the protocol on the input pair (X, Y). A *rectangle* R in M_f is a subset of entries in the function matrix such that $R = U \times V$ for some $U \subseteq \mathcal{X}$ and $V \subseteq \mathcal{Y}$.

Let $w_A(\cdot)$ and $w_B(\cdot)$ be non-negative weight functions defined on the row and column inputs, respectively, and extend it to subsets as well.[3] We say that a column input Y is α-*good* if $w_A(\{X \mid f(X, Y) = 1\}) \geqslant \alpha$.

Definition 2 ((α, β)-richness). A function matrix M_f is (α, β)-*rich* if

$$w_B(\{Y \mid Y \text{ is } \alpha\text{-good}\}) \geqslant \beta.$$

Let P be any deterministic $[a, b]$-protocol, i.e., Alice uses a bits and Bob uses b bits of communication. Let $f(X, Y)$ denote the output of P on the input pair (X, Y). It is well-known that P induces a partition of M_f into monochromatic rectangles, which we refer to as $[a, b]$-*partition* of M_f.

The deterministic version of the richness lemma [MNSW98] states that if a function f has deterministic protocol with low communication, then the function matrix has a heavy 1-monochromatic rectangle. We say that a rectangle $R = U \times V$ in M_f, not necessarily monochromatic, has *weight at least* $\alpha \times \beta$ if $w_A(U) \geqslant \alpha$ and $w_B(V) \geqslant \beta$.

Lemma 3 (Deterministic richness lemma, Miltersen et al. [MNSW98]). *If M_f is (α, β)-rich and f has a deterministic $[a, b]$-protocol, then there exists a 1-monochromatic rectangle in M_f that has weight at least $(\alpha/2^a) \times (\beta/2^{a+b})$.*

For randomized protocols, we will restrict ourselves to the case when the weight functions are probability distributions μ_A and μ_B on $\mathcal{X}$ and $\mathcal{Y}$, respectively. Let μ be the distribution on $\mathcal{X} \times \mathcal{Y}$ defined by $\mu(X, Y) = \mu_A(X) \cdot \mu_B(Y)$. The *weight* of any set of input pairs is the probability mass assigned to that set by μ. The randomized version of richness lemma states that if a function f has a randomized two-sided error protocol with low communication, and if the weight of the ones of f is large, then M_f has a heavy rectangle which is almost 1-monochromatic.

Lemma 4 (Randomized richness lemma, Miltersen et al. [MNSW98]). *Suppose function f has a randomized δ-error $[a, b]$-protocol, and the weight of the ones of f is at least c, where $c \geqslant 4\sqrt{\delta}$, then there exists a rectangle R in M_f of weight at least $(c/2^{a+2}) \times (c/2^{a+b+2})$ such that $\Pr_{(X,Y)\sim\mu}[f(X, Y) = 0 \mid (X, Y) \in R] \leqslant \sqrt{\delta}$.*

Note that Lemma 4 holds for two-sided error protocols. For one-sided error, meaning the protocol never errs on the zeros of f, the proof (Appendix A.2) shows that the rectangle R is in fact a 1-monochromatic rectangle for f.

[3] Given a weight function $w(\cdot)$ on a set T, and a subset $S \subseteq T$, $w(S) \stackrel{\text{def}}{=} \sum_{s \in S} w(s)$.

3. A cell–probe lower bound

In this section we show randomized cell–probe lower bounds for the following problem:

INTERSECT ALL: Let the database $\mathcal{D}$ be a family of n sets $Y_1, Y_2, \ldots, Y_n$ where every $Y_i \subseteq [d]$. Given a query $X \subseteq [d]$, does X intersect Y_i for every $1 \leqslant i \leqslant n$?

Let $0 < \varepsilon < 1$. We will prove a lower bound for INTERSECT ALL when d satisfies $(3200/\varepsilon^2)\log n \leqslant d \leqslant (10^5/(8\varepsilon^2)) \cdot n^{1-12\varepsilon}$. By Lemma 1, a lower bound for this problem in the cell–probe model would follow from a lower bound in the asymmetric communication complexity setting. We prove the latter by a rather straightforward application of Lemma 4. We first give an overview of this proof.

The function matrix for INTERSECT ALL has rows indexed by Alice's inputs which are all the sets $X \subseteq [d]$ and columns indexed by Bob's inputs which are all tuples $\langle Y_1, Y_2, \ldots, Y_n \rangle$ where $Y_i \subseteq [d]$ for every i. A crucial step in our proof is to restrict Alice's input to only some specific sets, namely, sets $X_i \in \mathcal{X}$ where $\mathcal{X}$ is a *design*. All sets $X \in \mathcal{X}$ have the same size $d' = \Omega(d)$ and the intersection of any two distinct sets $X, X' \in \mathcal{X}$ has size at most $\theta d'$ where $\theta > 0$ is a small constant. It is well-known that such designs with size $2^{\Omega(d)}$ exist and they have found many applications in complexity theory, most notably the Nisan–Wigderson's construction of a pseudorandom generator from hard functions [NW94].

Any two members X and X' of the design are almost disjoint. Therefore, when we pick a set Y (of some specific size) at random, the events $Y \cap X = \emptyset$ and $Y \cap X' = \emptyset$ are almost independent. This allows us to apply an inclusion–exclusion inequality to bound the probability that a randomly picked set Y intersects all the sets $X_1, X_2, \ldots, X_t$ for some $t \approx n$. Using this bound, we can immediately bound size of any 1-monochromatic rectangle in the matrix M_f.

Now we get to the details of the proof. We begin by defining a design.

Definition 5 (Design). A collection of sets $\{X_1, \ldots, X_m\}$ where $X_i \subseteq [d]$ is called a (d, m, α, β)-*design* if

1. $|X_i| = \lceil \alpha d \rceil, \ \forall i$,
2. $|X_i \cap X_j| \leqslant \beta d, \ \forall i \neq j$.

The following lemma shows the existence of designs (see, for instance, [NW94]). Appendix A.3 contains a proof of this lemma.

Lemma 6. *For every $\theta > 0$, there exist $c_1, c_2 > 0$ such that for all sufficiently large integers d, there exists a $(d, 2^{c_2 d}, c_1, \theta c_1)$-design. In fact, one can take $c_1 = \theta/2$ and $c_2 = \theta^2/96$.*

Fix any $\theta < 1/4$ for now; we will choose $\theta = \varepsilon/4$ later. Let $\mathcal{X}$ be the design guaranteed by the lemma, let

$$\mathcal{Y} \stackrel{\text{def}}{=} \{\langle Y_1, Y_2, \ldots, Y_n \rangle \mid Y_i \subseteq [d]\}$$

and let $d' \stackrel{\text{def}}{=} \lceil c_1 d \rceil$. $\mathcal{X}$ will be the set of inputs to Alice and $\mathcal{Y}$ will be the set of inputs for Bob. We will fix probability distributions $\mu_{\mathcal{X}}$ and $\mu_{\mathcal{Y}}$ on the inputs to Alice and Bob, respectively. (These will also play the role of the weight functions when applying the richness technique). $\mu_{\mathcal{X}}$ is the

uniform distribution on $\mathcal{X}$. $\mu_{\mathcal{Y}}$ is obtained by picking a tuple $\langle Y_1, Y_2, \ldots, Y_n \rangle$ where the sets $Y_i \subseteq [d]$ are picked independently and each set Y_i is picked by taking every $i \in [d]$ with probability q, where $q \stackrel{\text{def}}{=} (\ln n)/d'$. Note that in the above step, we assumed $d' \geqslant \ln n$ which follows since $d \geqslant (3200/\varepsilon^2)\log n$.

Let f denote the restriction of INTERSECT ALL to $\mathcal{X} \times \mathcal{Y}$. Recall that by definition, $M_f(X, \langle Y_1, \ldots, Y_n \rangle) = 1$ if and only if $X \cap Y_i \neq \emptyset$ for every i.

Lemma 7. *The weight of the ones of f is at least* $1/(2e)$.

Proof. Fix a set $X \in \mathcal{X}$, and let $Y = \langle Y_1, \ldots, Y_n \rangle$ be picked at random according to $\mu_{\mathcal{Y}}$. We have,

$$\Pr_{Y_i}[Y_i \cap X = \emptyset] = (1-q)^{|X|} = \left(1 - \frac{\ln n}{d'}\right)^{d'} \leqslant e^{-\ln n} = \frac{1}{n},$$

where the inequality follows from $1 - x \leqslant e^{-x}$. Therefore,

$$\Pr_Y[f(X, Y) = 1] = \Pr_{Y_1, \ldots, Y_n}[\forall i \;\; Y_i \cap X \neq \emptyset] \geqslant \left(1 - \frac{1}{n}\right)^n \geqslant \frac{1}{2e}. \qquad \square$$

Theorem 8. *Suppose* $R = U \times V$ *is a rectangle in* M_f *where* $|U| = n$ *and* $\mu_{\mathcal{Y}}(V) \geqslant \exp(-\frac{1}{2}n^{1-3\theta})$. *Then R is not* 1*-monochromatic*.

Proof. Let U consists of the row inputs $X_1, X_2, \ldots, X_n$. We will show that when a column $\langle Y_1, Y_2, \ldots, Y_n \rangle$ is picked at random according to distribution $\mu_{\mathcal{Y}}$, then the probability that the column has 1's in all n rows is at most $\exp(-\frac{1}{2}n^{1-3\theta})$. Thus we need to upper bound the probability of the event

$$Y_j \cap X_i \neq \emptyset, \quad \forall 1 \leqslant i, j \leqslant n.$$

Let us first compute the probability that a random set Y (where every element $i \in [d]$ is picked with probability q) intersects every X_i. Let E_i be the event that "$Y \cap X_i = \emptyset$". Clearly,

$$\Pr_Y[E_i] = (1-q)^{|X_i|} = \left(1 - \frac{\ln n}{d'}\right)^{d'} \geqslant \frac{1}{n^{1+\theta}},$$

where we used $1 - x \geqslant e^{-(1+\theta)x}$ if $\theta \leqslant 1/4$ and $x \leqslant 1/4$; since $d \geqslant (3200/\varepsilon^2)\log n$, the latter constraint is satisfied when applying the inequality. Furthermore, since for $i \neq j$, we have $|X_i \cap X_j| \leqslant \theta d'$, we have

$$\Pr_Y[E_i \wedge E_j] = (1-q)^{|X_i \cup X_j|} \leqslant \left(1 - \frac{\ln n}{d'}\right)^{(2-\theta)d'} \leqslant \frac{1}{n^{2-\theta}},$$

where in the last step we used $1 - x \leqslant e^{-x}$. Now we apply the following form of the *inclusion–exclusion* bound: if $E_1, \ldots, E_t$ are indicator random variables, then $\Pr[\bigvee_{i=1}^t E_i] \geqslant \sum_{i=1}^t \Pr[E_i] - \sum_{1 \leqslant i < j \leqslant t} \Pr[E_i \wedge E_j]$. Let $t = n^{1-2\theta}$.

$$\begin{aligned}
\Pr_Y\left[\bigwedge_{i=1}^{n}(Y\cap X_i\neq\emptyset)\right] &\leqslant \Pr_Y\left[\bigwedge_{i=1}^{t}(Y\cap X_i\neq\emptyset)\right]\\
&= 1-\Pr_Y\left[\bigvee_{i=1}^{t}(Y\cap X_i=\emptyset)\right]\\
&= 1-\Pr_Y\left[\bigvee_{i=1}^{t}E_i\right]\\
&\leqslant 1-\sum_{i=1}^{t}\Pr_Y[E_i]+\sum_{1\leqslant i<j\leqslant t}\Pr_Y[E_i\wedge E_j]\\
&\leqslant 1-t\cdot\frac{1}{n^{1+\theta}}+\frac{t^2}{2}\frac{1}{n^{2-\theta}}\\
&= 1-\frac{1}{n^{3\theta}}+\frac{1}{2}\frac{1}{n^{3\theta}}\\
&= 1-\frac{1}{2}\frac{1}{n^{3\theta}}.
\end{aligned}$$

Since Y_j, $1\leqslant j\leqslant n$ are picked independently, we have

$$\Pr_{Y_1,\ldots,Y_n}\left[\bigwedge_{j=1}^{n}\bigwedge_{i=1}^{n}Y_j\cap X_i\neq\emptyset\right]\leqslant\left(1-\frac{1}{2}\frac{1}{n^{3\theta}}\right)^n\leqslant\exp\left(-\frac{1}{2}n^{1-3\theta}\right),$$

where the last inequality follows by applying $1-x\leqslant e^{-x}$. □

Theorem 9. *Suppose $R=U\times V$ is a rectangle in M_f where $|U|=n$ and $\mu_Y(V)\geqslant 2w$, where $w=\exp(-\frac{1}{2}n^{1-3\theta})$. Then, $\Pr[f(X,Y)=0\mid(X,Y)\in R]\geqslant 1/(2n)$.*

Proof. This follows by carefully looking at the proof of Theorem 8. What we have shown is that the set of column inputs each of which has a 1 in all the n rows has weight at most w. Therefore, for a rectangle with n rows and columns with total weight $2w$, at least half the columns contain a zero in some row. This proves the theorem. □

Theorem 10. *For any $\varepsilon>0$ and for any two-sided error $[a,b]$-protocol for* Intersect All *with error $1/3$, either $a=\Omega(\varepsilon^2 d/\ln n)$ or $b=\Omega(n^{1-\varepsilon})$.*

Proof. This follows easily from Theorem 9 and Lemma 4. Assume on the contrary that there exists a protocol with error $1/3$ where Alice sends at most $10^{-5}\varepsilon^2 d/\ln n$ bits and Bob sends at most $n^{1-\varepsilon}$ bits. We first repeat the protocol $4\ln n$ times to get the error down to $1/n^4$. In this new protocol, Alice sends at most $10^{-5}4\varepsilon^2 d$ bits and Bob sends at most $4n^{1-\varepsilon}\ln n$ bits. By Lemma 7, the

ones in the matrix M_f have weight at least $1/(2e) \geqslant 1/8$. By Lemma 4, there exists a rectangle R with weight at least

$$\frac{1}{8 \cdot 2^{4\varepsilon^2 d/10^5+2}} \times \frac{1}{8 \cdot 2^{4\varepsilon^2 d/10^5+4n^{1-\varepsilon} \ln n+2}}$$

such that $\Pr_{(X,Y)\sim\mu}[f(X, Y) = 0 \mid (X, Y) \in R] \leqslant 1/n^2$. Let $\theta = \varepsilon/4$ so that the matrix M_f has $2^{c_2 d} = 2^{\theta^2 d/96}$ rows. Therefore the number of rows of the rectangle R is

$$2^{c_2 d}\frac{1}{8 \cdot 2^{4\varepsilon^2 d/10^5+2}} \geqslant 2^{\theta^2 d/96}\frac{1}{2^{\theta^2 d/200}} \geqslant 2^{\theta^2 d/200} \geqslant n,$$

where the first inequality follows since $c_2 = \theta^2/96$ (Lemma 6) and $\theta = \varepsilon/4$ and the third inequality follows from our assumption $d \geqslant (3200/\varepsilon^2)\log n$. The total weight of the column inputs of this rectangle is

$$\frac{1}{8 \cdot 2^{4\varepsilon^2 d/10^5+4n^{1-\varepsilon} \ln n+2}} \geqslant 2 \exp\left(-\frac{1}{2} n^{1-3\theta}\right),$$

which follows since $d \leqslant (10^5/(8\varepsilon^2)) \cdot n^{1-12\varepsilon}$. This contradicts Theorem 9 and we are done. □

Theorem 11. *For any $\varepsilon > 0$ and for any one-sided error $[a, b]$-protocol for* Intersect All *with error* $1/3$, *either $a = \Omega(\varepsilon^2 d)$ or $b = \Omega(n^{1-\varepsilon})$.*

Proof. This follows in a similar manner as the proof of Theorem 10. However, for protocols with one-sided error, Lemma 4 guarantees that the rectangle R is monochromatic. Thus we do not need to reduce the error probability and we save the $\ln n$ factor in communication. □

Corollary 12. *Any randomized algorithm for* Intersect All *in the cell–probe model that uses at most* $\mathrm{poly}(n, d)$ *cells with size* $\mathrm{poly}(d, \log n)$ *each, must make $\Omega(d/\log^2 n)$ probes. If the algorithm makes only one-sided error, it must make $\Omega(d/\log n)$ probes.*

Proof. Suppose there is a cell–probe algorithm with parameters s, b, t where $s = \mathrm{poly}(n, d)$ and $b = \mathrm{poly}(d, \log n)$. By Lemma 1, there is a communication protocol where Alice sends $t \log s$ bits and Bob sends tb bits and by Theorem 10, either $t \log s \geqslant \varepsilon^2 d/\log n$ or $tb \geqslant n^{1-\varepsilon}$. Thus, $t \geqslant \Omega(d/(\log n \cdot \log s))$ and since $\log s = O(\log n + \log d) = O(\log n)$, it follows that the number of probes $t \geqslant \Omega(d/\log^2 n)$. □

4. Applications

We discuss the applications of the above lower bound for partial match, exact nearest neighbor, and other related problems.

Partial Match: Given a database of n points in $\{0, 1\}^d$, and a query x in $\{0, 1, *\}^d$, is there a database point y such that for every i whenever $x_i \neq *$, we have $x_i = y_i$?

The partial match problem is equivalent to INTERSECT ALL in the following way. Given an instance of INTERSECT ALL with database $\mathcal{D} = \{Y_1, \ldots, Y_n\}$ and query X, where $Y_1, \ldots, Y_n, X \in \{0,1\}^d$, we obtain X' from X by replacing each 0 in X by a $*$ and each 1 by a 0. Now, it is easy to see that $(X', \mathcal{D})$ is a true instance for PARTIAL MATCH if and only if $(X, \mathcal{D})$ is a false instance for INTERSECT ALL. Therefore, using Lemma 1, Theorem 10, and the above reduction, we have

Theorem 13. *For any $\varepsilon > 0$, any two-sided error randomized algorithm in the cell–probe model for* PARTIAL MATCH *that makes t probes, either uses $2^{\Omega(d/(t \log^2 n))}$ cells or uses cells of size $\Omega(n^{1-\varepsilon}/t)$.*

In particular, if the number of cells is restricted to be $\mathrm{poly}(n, d)$ where each cell is of size $\mathrm{poly}(\log n, d)$, then the algorithm must make $\Omega(d/\log^2 n)$ probes.

SUBSET QUERY: Let U be a universe and let $\mathcal{P}$ be a family of subsets of U. Given a query set $Q \subseteq U$, is there a $P \in \mathcal{P}$ such that $Q \subseteq P$?

SUBSET QUERY is intimately related to PARTIAL MATCH. By virtue of the equivalence between these problems shown by Charikar et al. [CIP02], we obtain a cell–probe lower bound similar to Theorem 13 for the SUBSET QUERY problem as well.

Interpreting PARTIAL MATCH in a geometric manner as stated by Borodin et al. [BOR99], we obtain an improved cell–probe lower bound for the affine subspace problem: whether or not a query affine subspace contains at least one database point.

NEAREST NEIGHBOR: Let $\mathrm{ham}(\cdot,\cdot)$ denote the Hamming distance in $\{0,1\}^d$ and let $\mathcal{P}$ be a set of points in $\{0,1\}^d$. Given a query point $Q \in \{0,1\}^d$, find a point $P \in \mathcal{P}$ such that $\mathrm{ham}(P, Q) = \min_{P' \in \mathcal{P}} \mathrm{ham}(P', Q)$.

Using Theorem 10 and a reduction given by Borodin et al. [BOR99], we conclude that in the asymmetric communication problem for a decision version of the nearest neighbor problem, either Alice sends $\Omega(\varepsilon^2 d/\ln n)$ bits or Bob sends $\Omega(n^{1-\varepsilon})$ bits, for any $\varepsilon > 0$. The best previous bound for this problem were due to Barkol and Rabani [BR00] who showed that either Alice sends $\Omega(\varepsilon d)$ bits or Bob sends $\Omega(n^{\delta})$ bits, where $\delta < 1/8$ and ε depends on δ. Using other reductions stated by Borodin et al. [BOR99], our improved communication complexity lower bounds hold for decision versions of nearest neighbor in ℓ_p^d (d-dimensional real space under the ℓ_p norm), for all finite $p \geqslant 1$. Via easy reductions, the communication complexity lower bound of other geometric data structures problems is also improved. These include point location in an arrangement of hyperplanes, and a multi-dimensional generalization of the dictionary problem. For more details, see [BOR99].

ℓ_∞ c-NEAREST NEIGHBOR: Let ℓ_∞^d denote the d-dimensional real space under the ℓ_∞ norm and let $\mathcal{P}$ be a set of points in ℓ_∞^d. Given a query point $Q \in \ell_\infty^d$, is there a $P \in \mathcal{P}$ such that $|P - Q|_\infty \leqslant c \cdot \min_{P' \in \mathcal{P}} |P' - Q|_\infty$?

For $c < 3$ this problem is as hard as PARTIAL MATCH as shown by Indyk [Ind01]. Therefore, a near-optimal communication complexity lower bound and a cell–probe lower bound follows for the c-nearest neighbor in ℓ_∞ norm, for $c < 3$.

Via an easy reduction from this problem, we obtain similar bounds for the following *range search* problem as well: The database is a set of points in d-dimensional real space and given a

query which is a rectilinear range, does the query contain a database point. These bounds apply as well to the dual problem where the database is a set of ranges and the query is a point.

Appendix A. Proofs

A.1. Proof of Lemma 3

Proof. We use induction on $a+b$. If $a+b=0$ then $a=0, b=0$, the matrix consists solely of 1's and there is only one rectangle which is the whole matrix. So the lemma is clearly true in this case.

For the induction step, first consider the case when Bob sends the first bit. This splits the matrix M vertically into two parts, say M_1 and M_2. At least one of the two parts, say M_1, is $(\alpha, \beta/2)$-rich and it has $[a, b-1]$-partition. By induction hypothesis, one of the rectangles R in M_1 is 1-monochromatic and has weight at least $\frac{\alpha}{2^a} \times \frac{\beta/2}{2^{a+b-1}} = \frac{\alpha}{2^a} \times \frac{\beta}{2^{a+b}}$.

Now consider the case when Alice sends the first bit. This splits the matrix M horizontally into two parts, say M_1 and M_2. At least one of the two parts, say M_1 is $(\alpha/2, \beta/2)$-rich and has a $[a-1, b]$-partition. Using the induction hypothesis, there is a 1-monochromatic rectangle R with weight at least $\frac{\alpha/2}{2^{a-1}} \times \frac{\beta/2}{2^{a-1+b}} = \frac{\alpha}{2^a} \times \frac{\beta}{2^{a+b}}$. □

A.2. Proof of Lemma 4

Proof. Fix the random coins of the protocol such that the deterministic function g computed by the protocol satisfies

$$\Pr_{(X,Y)\sim\mu}[f(X,Y) \neq g(X,Y)] \leqslant \delta.$$

The ones in the matrix M_g have weight at least $c-\delta$ and it has an $[a,b]$-partition. Call a rectangle R in this partition *good* if

$$\Pr[f(X,Y) \neq g(X,Y) \mid (X,Y) \in R] \leqslant \sqrt{\delta}.$$

By an averaging argument,

$$\sum_{R:R \text{ is not good}} \mu(R) \leqslant \sqrt{\delta}.$$

Now, modify the protocol such that we output a 0 for all the inputs in bad rectangles. If g' is the function computed by this new protocol, note that $M_{g'}$ has an $[a,b]$ partition identical to that of M_g. The ones in the matrix $M_{g'}$ have weight at least $c-\delta-\sqrt{\delta} \geqslant c/2$, implying by another averaging argument that $M_{g'}$ is $(c/4, c/4)$-rich.

Applying Lemma 3, there is a 1-monochromatic rectangle R in the $[a,b]$ partition of $M_{g'}$ with weight at least $\frac{c}{2^{a+2}} \times \frac{c}{2^{a+b+2}}$. Since all the bad rectangles are 0-monochromatic, R must be a good rectangle. Moreover, M_g and $M_{g'}$ have identical values in R, therefore $g(X,Y)=1$ for

every $(X, Y) \in R$. By the definition of goodness,

$$\Pr_{(X,Y)\sim\mu}[f(X, Y) = 0 \mid (X, Y) \in R] \leqslant \sqrt{\delta}. \qquad \square$$

A.3. Proof of Lemma 6

Proof. We will prove Lemma 6 with $c_1 = \theta/2$ and $c_2 = \theta^2/96$. Thus we will construct a design of size $2^{\theta^2 d/96}$ where every set has size $\theta d/2$ and every pairwise intersection has size at most $\theta^2 d/2$. We will select sets $X_1, X_2, \ldots$ one by one in a greedy fashion. Having selected $X_1, X_2, \ldots, X_r$, if there exists a set $X, |X| = \theta d/2$ such that it intersects every X_i, $1 \leqslant i \leqslant r$ in at most $\theta^2 d/2$ points, we let $X_{r+1} = X$ and continue this process. Using a probabilistic argument, we show that as long as $r \leqslant 2^{\theta^2 d/96}$, there always exists such a set X.

Consider the process of picking a set X where each element of $[d]$ is selected with probability $3\theta/4$. We will use the following Chernoff bound (for example, see the book by Chazelle [Cha00, Lemma A.3]).

Lemma 14. *Suppose $Z_1, Z_2, \ldots, Z_t$ are independent* 0-1 *random variables and $\Pr[Z_i = 1] = p$ for every i. Let $Z = \sum_{i=1}^{t} Z_i$. Then for $0 < a < 1/2$,*

$$\Pr[Z > (1 + a)pt] \leqslant \exp(-pta^2/4),$$
$$\Pr[Z < (1 - a)pt] \leqslant \exp(-pta^2/2).$$

Using this bound with $p = 3\theta/4$, we have

$$\Pr[|X| < \theta d/2] < \exp(-\theta d/24) \leqslant 1/2,$$

and

$$\Pr[|X \cap X_i| > \theta^2 d/2] < \exp(-\theta^2 d/96).$$

Taking a union bound, it follows that as long as $r < 2^{\theta^2 d/96}$, there exists a set X of size at least $\theta d/2$ that intersects every X_i in at most $\theta^2 d/2$ points. Taking any subset of X with size $\theta d/2$ completes the proof. $\square$

References

[Ajt88] M. Ajtai, A lower bound for finding predecessors in Yao's cell probe model, Combinatorica 8 (1988) 235–247.

[BR00] O. Barkol, Y. Rabani, Tight bounds for nearest neighbor search and related problems in the cell probe model, in: Proceedings of the 32nd Annual ACM Symposium on the Theory of Computing, 2000, pp. 388–396.

[BV02] P. Beame, E. Vee, Time-space tradeoffs, multiparty communication complexity, and nearest-neighbor problems, in: Proceedings of the 34th Annual ACM Symposium on the Theory of Computing, 2002, pp. 688–697.

[BOR99] A. Borodin, R. Ostrovsky, Y. Rabani, Lower bounds for high dimensional nearest neighbor search and related problems, in: Proceedings of the 31st Annual ACM Symposium on the Theory of Computing, 1999, pp. 312–321.

[CCGL99] A. Chakrabarti, B. Chazelle, B. Gum, A. Lvov, A lower bound on the complexity of approximate nearest-neighbor searching on the Hamming cube, in: Proceedings of the 31st Annual ACM Symposium on the Theory of Computing, 1999, pp. 305–311.

[CR03] A. Chakrabarti, O. Regev, An optimal randomised cell probe lower bound for approximate nearest neighbor searching, Electronic Colloquium on Computational Complexity, 2003.

[CIP02] M. Charikar, P. Indyk, R. Panigrahy, New algorithms for subset query, partial match, orthogonal range searching, and related problems, in: Proceedings of the 29th International Colloquium on Algorithms, Logic, and Programming, 2002, pp. 451–462.

[Cha00] B. Chazelle, The Discrepancy Method: Randomness and Complexity, Cambridge University Press, Cambridge, 2000.

[Ind01] P. Indyk, On approximate nearest neighbors under ℓ_∞ norm, J. Comput. System Sci. 63 (4) (2001) 627–638.

[IM98] P. Indyk, R. Motwani, Approximate nearest neighbors: towards removing the curse of dimensionality, in: Proceedings of the 30th Annual ACM Symposium on the Theory of Computing, 1998, pp. 604–613.

[Kle97] J. Kleinberg, Two algorithms for nearest-neighbor search in high dimensions, in: Proceedings of the 29th Annual ACM Symposium on the Theory of Computing, 1997, pp. 599–608.

[Knu73] D. Knuth, The Art of Computer Programming: Sorting and Searching, Addison-Wesley, Reading, MA, 1973.

[KOR00] E. Kushilevitz, R. Ostrovsky, Y. Rabani, Efficient search for approximate nearest neighbor in high dimensional spaces, SIAM J. Comput. 30 (2) (2000) 451–474.

[Liu03] D. Liu, A strong lower bound for approximate nearest neighbor searching in the cell probe model, 2003, Submitted for publication.

[Mei93] S. Meiser, Point location in arrangement of hyperplanes, Inform. and Comput. 106 (2) (1993) 286–303.

[Mil94] P.B. Miltersen, Lower bounds for union-split-find related problems on random access machines, in: Proceedings of the 26th Annual ACM Symposium on the Theory of Computing, 1994, pp. 625–634.

[Mil99] P.B. Miltersen, Cell probe complexity—a survey, in: Pre-Conference Workshop on Advances in Data Structures at the 19th Conference on Foundations of Software Technology and Theoretical Computer Science, 1999.

[MNSW98] P.B. Miltersen, N. Nisan, S. Safra, A. Wigderson, On data structures and asymmetric communication complexity, J. Comput. System Sci. 57 (1) (1998) 37–49.

[NW94] N. Nisan, A. Wigderson, Hardness vs. randomness, J. Comput. System Sci. 49 (2) (1994) 149–167.

[Riv74] R. Rivest, Analysis of associative retrieval algorithms, Ph.D. Thesis, Stanford University, 1974.

[Riv76] R. Rivest, Partial match retrieval algorithms, SIAM J. Comput. 5 (1) (1976) 19–50.

[Yao81] A.C. Yao, Should tables be sorted, J. Assoc. Comput. Mach. 28 (3) (1981) 615–628.